| | Type | L # | Hits | Search Text | DBs | Time Stamp |
|---|---|---|---|---|---|---|
| 1 | BRS | L1 | 186 | replace near string | USPAT | 2004/11/05 10:58 |
| 2 | BRS | L2 | 8093 | s1 and (divider) | USPAT | 2004/11/05 09:36 |
| 3 | BRS | L3 | 7 | 1 and (divider) | USPAT | 2004/11/05 09:36 |
| 4 | BRS | L4 | 134 | modify near3 string | USPAT | 2004/11/05 10:57 |
| 5 | BRS | L5 | 47 | replace near string | EPO; JPO; DERWENT | 2004/11/05 10:58 |
| 6 | BRS | L6 | 11 | modify near3 string | EPO; JPO; DERWENT | 2004/11/05 10:58 |

```
### Status: Path 1 of [Dialog Information Services via Modem]

### Status: Initializing TCP/IP using (UseTelnetProto 1 ServiceID pto-dialog)
Trying 31060000009998...Open

DIALOG INFORMATION SERVICES
PLEASE LOGON:
 ******** HHHHHHHH SSSSSSSS?
### Status: Signing onto Dialog
 ********
ENTER PASSWORD:
 ******** HHHHHHHH SSSSSSSS? ********
Welcome to DIALOG
### Status: Connected


Dialog level 04.18.01D

Last logoff:  04nov04 13:54:22
Logon file405 05nov04 11:14:15
           *** ANNOUNCEMENT ***
                     ***
--Connect Time joins DialUnits as pricing options on Dialog.
See HELP CONNECT for information.
                     ***
--SourceOne patents are now delivered to your email inbox
as PDF replacing TIFF delivery. See HELP SOURCE1 for more
information.
                     ***
--Important Notice to Freelance Authors--
See HELP FREELANCE for more information
                     ***
NEW FILES RELEASED
***Beilstein Abstracts (File 393)
***Beilstein Facts (File 390)
***Beilstein Reactions (File 391)
***F-D-C Gold/Silver Sheet (File 184)
***BIOSIS Toxicology (File 157)
***IPA Toxicology (File 153)
                     ***
UPDATING RESUMED
                     ***
RELOADED
***Toxfile (File 156)

REMOVED
***Textile Technology Digest (File 119)
                     ***

     >>> Enter BEGIN HOMEBASE for Dialog Announcements <<<
     >>>    of new databases, price changes, etc.      <<<
                     ****
COREABS is set ON as an alias for 77,35,593,65,2,233,99,473,474,475.
COREFULL is set ON as an alias for 9,15,16,20,148,160,275,476,610,613,621,623,624,636,8
10,813.
SOFTFULL is set ON as an alias for 278,634,256.
EUROFULL is set ON as an alias for 348,349.
JAPOABS is set ON as an alias for 347.
HEALTHFULL is set ON as an alias for 442,149,43,444.
HEALTHABS is set ON as an alias for 5,73,151,155,34,434.
DRUGFULL is set ON as an alias for 455,129,130.
DRUGABS is set ON as an alias for 74,42.
INSURANCEFULL is set ON as an alias for 625,637.
INSURANCEABS is set ON as an alias for 169.
TRANSPORTFULL is set ON as an alias for 80,637.
TRANSPORTABS is set ON as an alias for 108,6,63.
ADVERTISINGFULL is set ON as an alias for 635,570,PAPERSMJ,PAPERSEU.
INVENTORYABS is set ON as an alias for 8,14,94,6,34,434,7.
```

```
BANKINGFULL is set ON as an alias for 625,268,626,267.
BANKINGABS is set ON as an alias for 139.
HEALTHALL is set ON as an alias for COREFULL,COREABS,HEALTHFULL,HEALTHABS.
INSURANCEALL is set ON as an alias for COREFULL,COREABS,INSURANCEFULL,INSURANCEABS.
RESERVATIONALL is set ON as an alias for COREFULL, COREABS.
OPERATIONSALL is set ON as an alias for COREFULL,COREABS,INVENTORYABS.
TRANSPORTALL is set ON as an alias for COREFULL,COREABS,TRANSPORTFULL,TRANSPORTABS.
ADVERTISINGALL is set ON as an alias for COREFULL,COREABS,ADVERTISINGFULL.
SHOPPINGALL is set ON as an alias for COREFULL,COREABS,ADVERTISINGALL,47.
INVENTORYALL is set ON as an alias for COREFULL,COREABS,INVENTORYFULL.
BANKINGALL is set ON as an alias for COREFULL,COREABS,BANKINGFULL,BANKINGABS.
PORTFOLIOALL is set ON as an alias for COREFULL,COREABS,BANKINGALL.
TRADINGALL is set ON as an alias for COREFULL,COREABS,BANKINGALL.
CREDITALL is set ON as an alias for COREFULL,COREABS,BANKINGALL.
FUNDSALL is set ON as an alias for COREFULL,COREABS,BANKINGALL,608.
* * *
SYSTEM:HOME
Cost is in DialUnits
Menu System II: D2 version 1.7.9 term=ASCII
                    *** DIALOG HOMEBASE(SM) Main Menu ***

 Information:
  1.  Announcements (new files, reloads, etc.)
  2.  Database, Rates, & Command Descriptions
  3.  Help in Choosing Databases for Your Topic
  4.  Customer Services (telephone assistance, training, seminars, etc.)
  5.  Product Descriptions

 Connections:
  6.  DIALOG(R) Document Delivery
  7.  Data Star(R)

    (c) 2003 Dialog, a Thomson business.      All rights reserved.

      /H = Help            /L = Logoff          /NOMENU = Command Mode


Enter an option number to view information or to connect to an online
 service.  Enter a BEGIN command plus a file number to search a database
(e.g., B1 for ERIC).
?b coreful, coreabs

>>>"COREFUL" is not a valid category or service name
>>>          77 does not exist
>>>1 of the specified files is not available
       05nov04 11:14:24 User242933 Session D190.1
            $0.00    0.220 DialUnits FileHomeBase
     $0.00  Estimated cost FileHomeBase
     $0.03  TELNET
     $0.03  Estimated cost this search
     $0.03  Estimated total session cost   0.220 DialUnits


SYSTEM:OS  - DIALOG OneSearch
  File  35:Dissertation Abs Online 1861-2004/Oct
        (c) 2004 ProQuest Info&Learning
  File 593:KOMPASS Central/Eastern Europe 2004/Jul
        (c) 2004 KOMPASS Intl.
  File  65:Inside Conferences 1993-2004/Oct W5
        (c) 2004 BLDSC all rts. reserv.
  File   2:INSPEC 1969-2004/Oct W4
        (c) 2004 Institution of Electrical Engineers
*File   2: Alert feature enhanced for multiple files, duplicates
removal, customized scheduling. See HELP ALERT.
  File 233:Internet & Personal Comp. Abs. 1981-2003/Sep
        (c) 2003 EBSCO Pub.
*File 233: File 233 is closed (no longer updating).
  File  99:Wilson Appl. Sci & Tech Abs 1983-2004/Sep
        (c) 2004 The HW Wilson Co.
```

```
  File 473:FINANCIAL TIMES ABSTRACTS 1998-2001/APR 02
        (c) 2001 THE NEW YORK TIMES
*File 473: This file will not update after March 31, 2001.
It will remain on Dialog as a closed file.
  File 474:New York Times Abs 1969-2004/Nov 04
        (c) 2004 The New York Times
  File 475:Wall Street Journal Abs 1973-2004/Nov 04
        (c) 2004 The New York Times

      Set  Items  Description
      ---  -----  -----------
?s (modify near string)
      S1       0  (MODIFY NEAR STRING)
?b corefull, coreabs
>>>          77 does not exist
>>>1 of the specified files is not available
       05nov04 11:15:02 User242933 Session D190.2
            $0.06    0.015 DialUnits File35
     $0.06  Estimated cost File35
            $0.19    0.030 DialUnits File593
     $0.19  Estimated cost File593
            $0.11    0.030 DialUnits File65
     $0.11  Estimated cost File65
            $0.23    0.030 DialUnits File2
     $0.23  Estimated cost File2
            $0.04    0.015 DialUnits File233
     $0.04  Estimated cost File233
            $0.07    0.030 DialUnits File99
     $0.07  Estimated cost File99
            $0.10    0.030 DialUnits File473
     $0.10  Estimated cost File473
            $0.05    0.015 DialUnits File474
     $0.05  Estimated cost File474
            $0.10    0.030 DialUnits File475
     $0.10  Estimated cost File475
            OneSearch, 9 files,  0.225 DialUnits FileOS
     $0.24  TELNET
     $1.19  Estimated cost this search
     $1.22  Estimated total session cost   0.445 DialUnits


SYSTEM:OS  - DIALOG OneSearch
  File   9:Business & Industry(R) Jul/1994-2004/Nov 04
        (c) 2004  The Gale Group
  File  15:ABI/Inform(R) 1971-2004/Nov 05
        (c) 2004 ProQuest Info&Learning
*File  15: Alert feature enhanced for multiple files, duplicate
removal, customized scheduling. See HELP ALERT.
  File  16:Gale Group PROMT(R) 1990-2004/Nov 05
        (c) 2004 The Gale Group
*File  16: Alert feature enhanced for multiple files, duplicate
removal, customized scheduling. See HELP ALERT.
  File  20:Dialog Global Reporter 1997-2004/Nov 05
        (c) 2004 The Dialog Corp.
  File 148:Gale Group Trade & Industry DB 1976-2004/Nov 05
        (c)2004 The Gale Group
*File 148: Alert feature enhanced for multiple files, duplicate
removal, customized scheduling. See HELP ALERT.
  File 160:Gale Group PROMT(R) 1972-1989
        (c) 1999 The Gale Group
  File 275:Gale Group Computer DB(TM) 1983-2004/Nov 05
        (c) 2004 The Gale Group
  File 476:Financial Times Fulltext 1982-2004/Nov 05
        (c) 2004 Financial Times Ltd
  File 610:Business Wire 1999-2004/Nov 01
        (c) 2004 Business Wire.
*File 610: File 610 now contains data from 3/99 forward.
Archive data (1986-2/99) is available in File 810.
  File 613:PR Newswire 1999-2004/Nov 05
```

(c) 2004 PR Newswire Association Inc
***File 613: File 613 now contains data from 5/99 forward.**
Archive data (1987-4/99) is available in File 813.
File 621:Gale Group New Prod.Annou.(R) 1985-2004/Nov 05
(c) 2004 The Gale Group
File 623:Business Week 1985-2004/Nov 05
(c) 2004 The McGraw-Hill Companies Inc
File 624:McGraw-Hill Publications 1985-2004/Nov 05
(c) 2004 McGraw-Hill Co. Inc
***File 624: Homeland Security & Defense and 9 Platt energy journals added**
Please see HELP NEWS624 for more
File 636:Gale Group Newsletter DB(TM) 1987-2004/Nov 05
(c) 2004 The Gale Group
File 810:Business Wire 1986-1999/Feb 28
(c) 1999 Business Wire
File 813:PR Newswire 1987-1999/Apr 30
(c) 1999 PR Newswire Association Inc
File 35:Dissertation Abs Online 1861-2004/Oct
(c) 2004 ProQuest Info&Learning
File 593:KOMPASS Central/Eastern Europe 2004/Jul
(c) 2004 KOMPASS Intl.
File 65:Inside Conferences 1993-2004/Oct W5
(c) 2004 BLDSC all rts. reserv.
File 2:INSPEC 1969-2004/Oct W4
(c) 2004 Institution of Electrical Engineers
***File 2: Alert feature enhanced for multiple files, duplicates**
removal, customized scheduling. See HELP ALERT.
File 233:Internet & Personal Comp. Abs. 1981-2003/Sep
(c) 2003 EBSCO Pub.
***File 233: File 233 is closed (no longer updating).**
File 99:Wilson Appl. Sci & Tech Abs 1983-2004/Sep
(c) 2004 The HW Wilson Co.
File 473:FINANCIAL TIMES ABSTRACTS 1998-2001/APR 02
(c) 2001 THE NEW YORK TIMES
***File 473: This file will not update after March 31, 2001.**
It will remain on Dialog as a closed file.
File 474:New York Times Abs 1969-2004/Nov 04
(c) 2004 The New York Times
File 475:Wall Street Journal Abs 1973-2004/Nov 04
(c) 2004 The New York Times

```
      Set  Items  Description
      ---  -----  -----------
?s (modify near string)
      S1       0  (MODIFY NEAR STRING)
?s modify (n) string
          286845  MODIFY
          366980  STRING
      S2      11  MODIFY (N) STRING
?type s2/3,ab/all
>>>No matching display code(s) found in file(s): 65, 593, 623-624, 810, 813
```

Consider all

**2/3,AB/1 (Item 1 from file: 9)**
DIALOG(R)File 9:Business & Industry(R)
(c) 2004 The Gale Group. All rts. reserv.

1555951 Supplier Number: 01555951
**GEC-Marconi To Update U.K.'s Sing Ray Torpedo**
**(Contract to extend the life of torpedoes valued at UKPd109 mil ($168.8 mil))**
Defense News, v 11, n 30, p 34
July 29, 1996
DOCUMENT TYPE: Journal; News Brief ISSN: 0884-139X (United States)
LANGUAGE: English RECORD TYPE: Abstract

ABSTRACT:
GEC-Marconi Ltd (Stanmore, UK) has been given a UKPd109 mil ($168.8 mil) contract to **modify String** Ray torpedoes for the British Royal Navy and Royal Air Force. The improvements are to extend the useful life of the

torpedoes and to enhance its weapons performance.

**2/3,AB/2 (Item 1 from file: 160)**
DIALOG(R)File 160:Gale Group PROMT(R)


01829005
**TRACE TECHNOLOGY - NEW PRODUCT**
News Release November 11, 1987 p. 1

Trace Technology have launched an EPROM Emulator which has facilities normally only seen in processor emulators. No Interface Card is required, simply plug into the RS232 Port of your Computer. Power is provided via a plug-in power transformer. The trace Emulator E102 replaces the EPROM in a target system and via RS232 Data transfer, can accept data in Intel Hex, Motorola and TEX HEX formats. The RS232 port enables this emulator to be used with almost any PC from IBM's and compatible to BBC micro's and other home computers. The E102 emulates most commonly available EPROMS from 2732 up to 27512 as well as some of the more dated types such as 2532 and 2564. The Trace Emulator allows access to processosr control signals and the user define a window in the EPROM space which can be "written to" as if it were system RAM. This makes the E102 a powerful debugging tool without having to pay the price of a processor emulator. This facility overcomes one inherent limitation of most EPROM emulators, allowing information to be passed from the target system to the host computer. In addition the Trace Emulator has a host of facilities including - memory dump, memory **modify** , **string** search, copy, automatic baud rate selection and an extensive HELP Menu. The target system can be reset from the host computer, which avoids having to turn power off and on to get the code to run. The E102 has been designed with the engineer in mind. It is a functional tool capable of expansion in the future. The current model can be expanded to emulate up to 4 EPROMS and will also accept the forthcoming breakpoint card with trace memory facility.

Full text available on PTS New Product Announcements.

**2/3,AB/3 (Item 1 from file: 275)**
DIALOG(R)File 275:Gale Group Computer DB(TM)


02084680 SUPPLIER NUMBER: 19609757 (USE FORMAT 7 OR 9 FOR FULL TEXT)
**Ask our advisors (News Briefs)**
PC Magazine, v16, n14, p361(3)
August, 1997
ISSN: 0888-8507 LANGUAGE: English RECORD TYPE: Fulltext
WORD COUNT: 3219 LINE COUNT: 00237

**2/3,AB/4 (Item 2 from file: 275)**
DIALOG(R)File 275:Gale Group Computer DB(TM)


01830920 SUPPLIER NUMBER: 17269835 (USE FORMAT 7 OR 9 FOR FULL TEXT)
**Add client-side security to your applications. (a user object for PowerBuilder 4.0 that provides client security for applications)(Tutorial)**
Horwith, Michael
Data Based Advisor, v13, n7, p124(3)
August, 1995
DOCUMENT TYPE: Tutorial ISSN: 0740-5200 LANGUAGE: English
RECORD TYPE: Fulltext; Abstract
WORD COUNT: 1328 LINE COUNT: 00117

ABSTRACT: The most popular of the two kinds of client/server security is database server rights, and all major RDBMS vendors provide data control language extensions for their SQL dialects that limit or grant rights to data. Application security is more difficult in the client, however, because there is no built-in security scheme to apply to PowerBuilder

programs. A security object is presented that can be used to limit access to menu choices and DataWindows throughout a PowerBuilder application. The object is a custom class that supports security for menus, DataWindow controls, or DataWindow object columns; security can be set by entering the security code into the tag attribute of any of these components. The security object is declared as a global variable in the client code; it has one instance variable that is a string, and the object can be extended to carry around the user name and user ID.

**2/3,AB/5 (Item 3 from file: 275)**
DIALOG(R)File 275:Gale Group Computer DB(TM)


01713571 SUPPLIER NUMBER: 15102641 (USE FORMAT 7 OR 9 FOR FULL TEXT)
**An easy way to build a calendar. (Client/Server Advisor: PowerBuilder Pro)(Column) (Tutorial)**
Horwith, Michael
Data Based Advisor, v12, n4, p116(5)
April, 1994
DOCUMENT TYPE: Tutorial ISSN: 0740-5200 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1071 LINE COUNT: 00153

ABSTRACT: Datawindow-based calendars are superior to those built with standard controls on a window. There are two basic parts involved in developing a pop-up calendar in PowerBuilder database application development software: the datawindow object and the window containing the object's associated datawindow control. The datawindow object has a freeform presentation style and an external data source. The object lacks data and is comprised entirely of static text controls. The dwModify and dwDescribe functions make the described calendar a valuable asset in the developer's PowerBuilder base library. Detailed instructions for developing the calendar are provided.

**2/3,AB/6 (Item 4 from file: 275)**
DIALOG(R)File 275:Gale Group Computer DB(TM)


01380215 SUPPLIER NUMBER: 08840672 (USE FORMAT 7 OR 9 FOR FULL TEXT)
**Managing a natural resource. (managing the records of whale movements)**
Saunders, Kimberly M.
Data Based Advisor, v8, n9, p64(4)
Sept, 1990
ISSN: 0740-5200 LANGUAGE: ENGLISH RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 2410 LINE COUNT: 00187

ABSTRACT: Keeping track of the tail fin photos of humpback whales was solved with the R:BASE database management system from Microrim Inc and a video disc player. The National Marine Mammal Laboratory (NMML) uses the system to manage over 9,000 photographs on file. A MATCH routine is used to compare new photos of whale sightings to old photos of the same whales. It helps scientists to track the movement of the humpback whale population.

**2/3,AB/7 (Item 1 from file: 621)**
DIALOG(R)File 621:Gale Group New Prod.Annou.(R)


01057191 Supplier Number: 40212105
**TRACE TECHNOLOGY - NEW PRODUCT**
News Release, p1
Nov 11, 1987
Language: English Record Type: Fulltext
Document Type: Magazine/Journal; Trade
Word Count: 303

**2/3,AB/8 (Item 2 from file: 621)**
DIALOG(R)File 621:Gale Group New Prod.Annou.(R)


01057190 Supplier Number: 40212104
**Trace Technology have launched an EPROM Emulator which has facilities normally only seen in processor emulators.**
News Release, p1
Nov 11, 1987
Language: English Record Type: Fulltext
Document Type: Magazine/Journal; Trade
Word Count: 300

**2/3,AB/9 (Item 1 from file: 2)**
DIALOG(R)File 2:INSPEC


7843144 INSPEC Abstract Number: A2004-05-1117-002
**Title: The euler characteristic and the first chern number in the covariant phase space formulation of string theory**
Author(s): Cartas-Fuentevilla, R.
Author Affiliation: Inst. de Fisica, Univ. Autonoma de Puebla, Puebla Pue, Mexico
Journal: Journal of Mathematical Physics vol.45, no.2 p.602-8
Publisher: AIP,
Publication Date: Feb. 2004 Country of Publication: USA
CODEN: JMAPAQ ISSN: 0022-2488
SICI: 0022-2488(200402)45:2L.602:ECFC;1-O
Material Identity Number: J090-2004-001
U.S. Copyright Clearance Center Code: 0022-2488/2004/45(2)/602/7/$22.00
Language: English
Abstract: Using a covariant description of the geometry of deformations for extendons, it is shown that the topological corrections for the string action associated with the Euler characteristic and the first Chern number of the normal bundle of the world sheet, although do not give dynamics to the **string** , **modify** the symplectic properties of the covariant phase space of the theory. Future extensions of the present results are outlined.
Subfile: A


**2/3,AB/10 (Item 2 from file: 2)**
DIALOG(R)File 2:INSPEC


5175005 INSPEC Abstract Number: A9605-9760L-002
**Title: Correlations between black holes formed in cosmic string breaking**
Author(s): Emparan, R.
Author Affiliation: Dept. de Fisica de la Mater. Condensada, Pais Vasco Univ., Bilbao, Spain
Journal: Physical Review D (Particles, Fields, Gravitation, and Cosmology) vol.52, no.12 p.6976-81
Publisher: AIP for APS,
Publication Date: 15 Dec. 1995 Country of Publication: USA
CODEN: PRVDAQ ISSN: 0556-2821
SICI: 0556-2821(19951215)52:12L.6976:CBBH;1-G
Material Identity Number: O960-96001
U.S. Copyright Clearance Center Code: 0556-2821/95/52(12)/6976(6)/$06.00
Language: English
Abstract: An analysis of cosmic string breaking with the formation of black holes attached to the ends reveals a remarkable feature: the black holes can be correlated or uncorrelated. We find that, as a consequence, the number-of-states enhancement factor in the action governing the formation of uncorrelated black holes is twice the one for a correlated pair. We argue that when an uncorrelated pair forms at the ends of the string, the physics involved is more analogous to thermal nucleation than to particle-antiparticle creation. Also, we analyze the process of intercommuting strings induced by black hole annihilation and merging.

Finally, we discuss the consequences for grand unified strings. The process whereby uncorrelated black holes are formed yields a rate which significantly improves over those previously considered, but still not enough to **modify string** cosmology.
Subfile: A


**2/3,AB/11 (Item 3 from file: 2)**
DIALOG(R)File 2:INSPEC


02503263 INSPEC Abstract Number: C85041841
**Title: Utility program-memory file management for your 'pocket'**
Author(s): Ernotte, D.
Journal: Micro Systemes no.54 p.183-6
Publication Date: June 1985 Country of Publication: France
CODEN: MSYSDT ISSN: 0183-5084
Language: French
Abstract: A memory file management program is described and relevant listings given for the loading in BASIC and with codings listed in decimal for the software to simulate the management of a RAM diskette enabling one to load, execute, **modify** , **string** and kill programs at will. Various aspects of the program are described in detail.
Subfile: C
?type s2/3,9/5

**2/9/5 (Item 3 from file: 275)**
DIALOG(R)File 275:Gale Group Computer DB(TM)


01713571 SUPPLIER NUMBER: 15102641 (THIS IS THE FULL TEXT)
**An easy way to build a calendar. (Client/Server Advisor: PowerBuilder Pro)(Column) (Tutorial)**
Horwith, Michael
Data Based Advisor, v12, n4, p116(5)
April, 1994
DOCUMENT TYPE: Tutorial ISSN: 0740-5200 LANGUAGE: ENGLISH
RECORD TYPE: FULLTEXT; ABSTRACT
WORD COUNT: 1071 LINE COUNT: 00153

ABSTRACT: Datawindow-based calendars are superior to those built with standard controls on a window. There are two basic parts involved in developing a pop-up calendar in PowerBuilder database application development software: the datawindow object and the window containing the object's associated datawindow control. The datawindow object has a freeform presentation style and an external data source. The object lacks data and is comprised entirely of static text controls. The dwModify and dwDescribe functions make the described calendar a valuable asset in the developer's PowerBuilder base library. Detailed instructions for developing the calendar are provided.
TEXT:
Here's how to build a calendar in PowerBuilder using a datawindow. You can build it with standard controls on a window, but it's too slow. A datawindow-based calendar is superior.
There are two major parts to developing a popup calendar in PowerBuilder using a datawindow--a datawindow object, and a window that contains the datawindow control associated with that object.
Creating a datawindow object
The datawindow object has a freeform presentation style, with an external data source. There isn't any data in the object--the datawindow object is made up of static text controls. First, create a control for the header which contains the month and year, seven controls for the days of the week, and a 7x6 matrix of 42 controls for the days in the month. These controls should be called st_header, st_sunday through st_saturday, and st_01 through st_42. The controls are named sequentially from 1 to 42 in calendar fashion - st_01 through st_07 on row one, st_08 through st_14 on row two, etc. The calendar can be placed in the header band of the datawindow painter. The only difficult part of building the calendar is

aligning and spacing the controls. The arrow keys can be used for fine movements of selected objects.

Displaying the calendar

Now that we have a calendar datawindow object, the next step is to build a window to display the calendar. The window, which we'll call w_f_calendar (The "f" in the name designates the window as an object in our base class library), should contain a datawindow control and some command buttons. The VCR command buttons on the calendar header are used to increment or decrement the calendar by months or years. The OK button is used to accept a chosen date; Cancel is used to close the window without accepting a date. The OK and Cancel buttons are appropriate. This window acts as a base class for inherited windows that customize it.

The first job of our window is to draw the calendar with the appropriate month displayed. If the user double clicks on an empty date field, the calendar starts with system date; otherwise, it starts with the date in the clicked field. In order to find the appropriate start date in the response window, we use a global variable. In our library, global variables are contained in a structure, so the variable containing the date is selected in the open event of the calendar window. An instance variable (iVar) in the calendar window is then updated. Here is the calendar window open event:

```
string ls color
// set the iVar id_activedate.  Start with
// today if no date seeded
IF Year (gst_app_info.d_cal_start) = -1 THEN
id_activedate = today ()
ELSE
id_activedate = gst_app_info.d_cal_start
END IF
// initialize startdate iVar to first day of
// active date's month
id_startdate = Date ( Year (id_activedate), &
Month (id_activedate), 1 )

// set first time iVar, used to cause current
// date to be highlighted
ib_firsttime = true

// set iVar to say whether a date is true
ib_dateok = true

  // set iVar for unselected backgroundcolor
ls_color = dw_calendar.dwDescribe( &
"st_01.background.color" )
il_unselectedcolor = Long( ls_color )

  // default the selected color to cyan
il_selectedcolor = RGB ( 0, 255,255 )

  // call user event drawcalendar
Postevent (this, "drawcalendar" )
 [TABULAR DATA OMITTED]
```

The last thing that occurs in the open event is a call to the drawcalendar event. This is a user-defined event that is mapped to the event_id pbm_custom01, and this is where the calendar static text objects st_01 through st_42 and the header st_header are populated.

```
int i  // loop counter
int li_Start  // loop start variable
int li_End  // loop end variable
int li_First  // array element, 1st of month
int li_DaysInMonth  // number of days n month
int li_Day  // day of month
int li_Active  // array element of active date
string ls_objectname  // date object name
string ls_today  // helps construct ls_objectname

string ls_modify  // dwModify string
string ls_header  // dwModify string for header
```

```
// set title for calendar
ls_header = String (id_startdate, "mmmm") + ", " &
+ String (Year (id_startdate), "####")

// What array element is first of the month?
li_first = DayNumber (Date (Year (id_startdate), &
Month (id_startdate) , 1))

// Set the iVar ii_startele
ii_startele = li_first

// Fill iVar array of text of days
// to day 1 with empty string
li_End = li_First - 1
FOR i = 1 TO li_End
is_a_days [i] = ""
NEXT

// calculate number of days in month
li_DaysInMonth = gf_f_daysInMonth ( id_startdate )

// Fill in days in month
li_Start = li_End + 1
li_Active = li_Start + Day ( id_activedate ) - 1
li_End = li_Start + li_DaysInMonth - 1
li_Day = 0
FOR i = li_Start TO li_End
li_Day = li_Day + 1
is_a_days [i] = String ( li_Day, "[General]" ) &

NEXT
// fill in the rest of the month with blanks
li_Start = li_End + 1
FOR i = li_Start TO 42
is_a_days[i] = ""
NEXT

// update datawindow
DwModify ( dw_calendar, &
'st_header.text = "'+ls_header+'" ' + &

'st_01.text = "' + is_a_days[1] + '" + &
'st_02.text = "' + is_a_days[2] + '"' + &
'st_03.text = "' + is_a_days[3] + '"' + &
'st_04.text = "' + is_a_days[4] + '"' + &
'st_05.text = "' + is_a_days[5] + '"' + &
'st_06.text = "' + is_a_days[6] + '"' + &
'st_07.text = "' + is_a_days[7] + '"' + &
'st_08.text = "' + is_a_days[8] + '"' + &
'st_09.text = "' + is_a_days[9] + '"' + &
'st_10.text = "' + is_a_days[10] + '"'+ &
'st_11.text = "' + is_a_days[11] + '"'+ &
'st_12.text = "' + is_a_days[12] + '"'+ &
'st_13.text = "' + is_a_days[13] + '"'+ &
'st_14.text = "' + is_a_days[14] + '"'+ &
'st_15.text = "' + is_a_days[15] + '"'+ &
'st_16.text = "' + is_a_days[16] + '"'+ &
'st_17.text = "' + is_a_days[17] + '"'+ &
'st_18.text = "' + is_a_days[18] + '"'+ &
'st_19.text = "' + is_a_days[19] + '"'+ &
'st_20.text = "' + is_a_days[20] + '"'+ &
'st_21.text = "' + is_a_days[21] + '"'+ &
'st_22.text = "' + is_a_days[22] + '"'+ &
'st_23.text = "' + is_a_days[23] + '"'+ &
'st_24.text = "' + is_a_days[24] + '"'+ &
'st_25.text = "' + is_a_days[25] + '"'+ &
'st_263text = "' + is_a_days[26] + '"'+ &
'st_27.text = "' + is_a_days[27] + '"'+ &
'st_28.text = "' + is_a_days[28] + '"'+ &
```

```
'st_29.text = "' + is_a_days[29] + '"'+ &
'st_30.text = "' + is_a_days[30] + '"'+ &
'st_31.text = "' + is_a_days[31] + '"'+ &
'st_32.text = "' + is_a_days[32] + '"'+ &
'st_33.text = "' + is_a_days[33] + '"'+ &
'st_34.text = "' + is_a_days[34] + '"'+ &
'st_35.text = "' + is_a_days[35] + '"'+ &
'st_36.text = "' + is_a_days[36] + '"'+ &
'st_3X.text = "' + is_a_days[37] + '"'+ &
'st_38.text = "' + is_a_days[38] + '"'+ &
'st_39.text = "' + is_a_days[39] + '"'+ &
'st_40.text = "' + is_a_days[40] + '"'+ &
'st_41.text = "' + is_a_days[41] + '"'+ &
'st_42.text = "' + is_a_days[42] + '"'+ &

// First time so highlight today...
IF ib_firsttime THEN
ib_firsttime = FALSE

// update instance var
ii_activeele = li_Active

// calculate objectname for today ...
IF li_Active [lesser than] 10 TEN
ls_Today = "0" + String ( li_Active. "##" )
ELSE
ls_Today = String ( li_Active, "##" )
END IF
ls_objectname = "st_"+ls_today

// build dwmodigy string to change color
ls_moify = Left ( ls_objectname, 5 ) + &
'.background.color = "' + &
String (il_selectedcolor) +'" ' + &
Left ( ls_objectname, 5 ) + &
'.background.mode = "0"'

dwModify dw_calendar, ls_modify)
END IF
```

The drawcalendar event calls a user-defined function to calculate the numbers of days in a month. It constructs the first day of the next month and subtracts one day. The variable ad_date is an argument of any date in the month.

```
date ld part 1
integer li_DaysInMonth

// construct last day of previous month
ld_part1 = RelativeDate (ad_date, -Day( ad_date))

//add 35 days to last day of previous month,
//substract day of the month from
//constructed date.
li_DaysInMonth = Day (RelativeDate(&
RelativeDate(ld_part1, 35 ),&
-(Day(RelativeDate(ld_part1, 35)))))
RETURN li_DaysInMonth
```

Now in the calendar is drawn. The VCR buttons on the calendar header are used to increment and decrement months and years. Here is the code to increment the calendar by a month. In the clicked event of the > button:

```
string ls_oldobjectname  // old static text object
string ls_ modify   //  string  to modify
//  set the hourglass
SetPointer ( hourglass! )
// construct the active static text
// object name from the iVar ii_activeele
IF ii_activeele [lesser than] 10 THEN
ls_oldobjectname = st_0" &
+ String ( ii_activeele,#")
ELSE
```

```
  lsoldobjectname = "st_" &
+ String( ii_activeele, "##")
END IF

  // turn off the old highlighting
ls_modify = Left ( ls_oldobjectname, 5 ) &
+ '.background.color " "' &
+ String (il_unselectedcolor) + '"' &
+ Left ( ls_oldobjectname, 5 ) &
+ '.background.mode = "0"'

  dwModify ( dw_calendar, ls_modify)

  // calculate first of next month
id_stardate = wf_firstofnextmont (id_startdate)
// calculate new active date
id_activedate = gf_F_addmonth ( id_activedate )
// tell calendar to reset selected date
ib_firsttime = true
// draw the calendar
TriggerEvent ( parent, "drawncalendar" )
```

The key to this code is updating the instance variables that represent the start of the month, the selected date, and the flag to highlight the selected date. This has to be done because the same day of the month a month later will not fall in the same array element, so a different static text object will have to be highlighted. In addition, the current day may not exist in the following month, given the different number of days in a month. The other increment and decrement command buttons are basically the same, with an obvious difference in the calculation of id_startdate and id_activedate. The code to add and subtract months and years is basically the same in any language.

Pulling it together

We need to tie this window to the client code that contains a datawindow with a date field. When the date field is doubleclicked, the calendar should appear. To accomplish this, use a user object datawindow control, rather than a standard datawindow control, when placing the control on the client code window. The user object datawindow control has the following code in the doubleclicked event:

```
// what column did the user dclick on?
ls_objectname = this.DWGetObjectAtPointer()
ls_objectname = Left ( ls_name, &
Pos ( ls_objectname, "[unkeyable]t" )-1)

// is it a date column?
ls_syntax = ls_objectname+". Coltype )
// if it is a date or a datetime ...
IF ls_datatype = "date" OR ls_datatype = "datetime"
THEN

// get date value, put in a global
// variable checked by calendar window
// open event
IF ls_datatype = "date"  THEN
gst_app_info.d_cal_start = this.GetItemDate( &
this.GetRow(), ls_objectname )
ELSE
gst_app_info.d_cal_start = Date(&
this.GetItemDateTime( &
this.GetRow(), ls_objectname ) )
END IF

// open the calendar
open (w_f_calendar)
END IF
```

The last detail is to replace the date if the calendar is closed with the OK button. The clicked event of the OK button reads:

```
dataWindow ldw
// Get active datawindow from
```

```
     // global structure
     ldw = gst_app_info.dw_current
     // update datawindow with new date.  Convert
     // to date or datetime datatype
     IF ldw.dwDescribe( ldw.getColumnname() &
     + ".ColType" ) = "date" THEN
     ldw.SetItem( ldw.GetRow(), ldw.GetColumn(), &
     id_activedate )
     ELSE
     ldw.SetItem( ldw.GetRow(), ldw.GetColumn(), &
     END IF
     // close the calendar window
     Close ( parent )
      Conclusion
      The dwModigy and dwDescribe function makes this calendar to realistic
tool to add your PowerBuilder base library.
          COPYRIGHT 1994 Data Based Solutions Inc.

 SPECIAL FEATURES:  illustration; table; program
 DESCRIPTORS:  Calendars; Programming Instruction; Tutorial; Application
  Development Software; Database
 SIC CODES:  7372  Prepackaged software
 TRADE NAMES:  PowerBuilder (Database application development software)--
  Usage
 FILE SEGMENT:  CD File 275
?
```